Remarks

Reconsideration of the above-caption application is respectfully requested. All previously pending claims (1-23) have been rejected as being obvious over Fayyad et al. '708 in view of Tendick.

To overcome the rejections, Claim 1 has been amended to recite using a distribution of the perturbed data to generate an estimate of a distribution of the original data, and using the estimate of the distribution of the original data to generate a data mining model. Support for this amendment can be found on page 7, line 21 continuing to page 8, line 1 and in Figure 4. Claim 7 has been amended to recite that the server does not have access to the original values, as disclosed on page 7, line 14. Claim 21 has been amended to correct an informality in verb tense, and Claims 14-19 have been canceled. Claims 1-13 and 15-23, of which Claims 1, 7, and 13 are independent, remain pending.

Rejections Under 35 U.S.C. §103

The claims have been rejected under 35 U.S.C. §103 as being unpatentable over Fayyad et al. '708 in view of Tendick, with the secondary reference being used as a teaching of maintaining privacy. Fayyad et al. is directed only to using a perturbed value of a ***mean of original values***, not the original values themselves, to find a starting centroid for candidate data clusters, col. 2, lines 4-5, that is "cheaper" than doing a full-blown cluster operation, col. 5, lines 61-65. Fayyad et al. does not use perturbed values of original values at all. Instead, once the starting centroids are found, Fayyad et al. use actual original data to generate the models.

invention

p. Art.

I

With this understanding of Fayyad et al. in mind, attention is directed to the present claims, starting with Claim 1. As now amended, Claim 1 recites using a distribution of perturbed values to generate an estimate of the original distribution, and then using this estimate to generate the model. Fayyad et al. seems to be completely silent on this previously unclaimed feature.

Claim 7 explicitly requires randomizing, at a user computer, original values of numeric attributes to render perturbed values, and then sending the perturbed values to a server computer not having access to the original data for processing the perturbed values to generate a model. The examiner has taken the position that the perturbation of the ***mean*** of the centroid discussed at col. 2, lines 32-36 of Fayyad et al. reads on perturbing original data. He has further taken the position that the discussion of the computer system peripherals at col. 4, lines 37-41 reads on generating perturbed values at a user computer and processing them at a server.

As now amended, Claim 7 requires that the server not have access to original data. Whatever the merits of the examiner's comment that Fayyad et al. teaches a user and server, it is clear that the computer processing data in Fayyad et al. indeed has access to original data. Thus, this rejection has been overcome.

Additionally, Applicant respectfully disagrees with the examiner's positions summarized above. First, the mean of a set of original values is not the same thing as the original values themselves; rather, it is a statistical representation of the original values. Consequently, perturbing a mean value as Fayyad et al. does is not the same thing as perturbing original values, in contrast to what is recited in Claim 7. Instead, Fayyad et al. at most can be said to perturb a statistical representation of original values. Since Fayyad et al. nowhere considers privacy, but only finding a good starting point for cluster centroids, there is absolutely no reason for Fayyad et al. to suggest or be modified to perturb anything other than a statistic, and not original values.

Second, Applicant believes that the relied-upon portion of Fayyad et al. alleged to teach generating perturbed values at a user computer and sending them on to a server for processing is not quite accurate. While Fayyad et al. contains a large volume of boilerplate about computer peripherals, and the fact that there are such things as computer networks (see col. 5, lines 4-30), nothing in Fayyad et al. suggests generating the perturbed values at a user computer, but processing them on a server computer. There is simply no reason to do so since Fayyad et al. is not directed to maintaining privacy. In any case, wherever the perturbed data is generated in Fayyad et al., there is no teaching or suggestion that the original data remain unavailable to the computer that generates the model. If the original data were not available, the invention of Fayyad et al. wouldn't work, see MPEP §2143.01 (citing In re Gordon).

With respect to dependent Claim 12, the examiner alleges that Fayyad et al. col. 10, lines 18-25 teaches perturbing categorical values of categorical attributes by selectively replacing the categorical values with other values based on a probability. The relied-upon section of Fayyad et al., however, nowhere mentions the concept of "categorical attributes". In fact, the entire patent doesn't mention the concept. Fayyad et al. appears to exclusively consider numerical attributes. In any case, there is no mention at all of replacing categorical values with other values based on a probability.

Claim 13 requires receiving perturbed values from the user computers, with the perturbed values representing randomized versions of the original values, and then generating a classification model using the perturbed values and not using the original values. As discussed above, Fayyad et al. nowhere teaches or suggests ***not*** using original values in generating the cluster model. Interestingly, this last limitation of Claim 13 was not mentioned in the rejection.

The Examiner is cordially invited to telephone the undersigned at (619) 338-8075 for any reason which would advance the instant application to allowance.

Respectfully submitted,

John L. Rogitz
Registration No. 33,549
Attorney of Record
750 B Street, Suite 3120
San Diego, CA 92101
Telephone: (619) 338-8075

JLR:jg

MARKED UP VERSION SHOWING CHANGES

1. (amended) A computer-implemented method for obtaining data from at least one user computer via the Internet while maintaining the privacy of a user of the computer, comprising the acts of:

perturbing original data associated with the user computer to render perturbed data; [and]

using a distribution of the perturbed data, generating at least one estimate of a distribution of the original data; and

using the estimate of the distribution of the original data, generating at least one data mining model.

7. (amended) A computer system including a program of instructions including structure to undertake method acts comprising:

at a user computer, randomizing at least some original values of at least some numeric attributes to render perturbed values;

sending the perturbed values to a server computer not having access to the original values; and

at the server computer, processing the perturbed values to generate at least one classification model.

21. (amended) The method of Claim 20, wherein the user computer uses [used] the model on original data to render a classification, and then sends the classification to the Web site.

distance between z and w_i (or between a and w_i) is approximated to be the distance between the midpoints of the intervals in which they lie. Also, the density function $f_X(a)$ is approximated to be the average of the density function in the interval in which the attribute "a" lies.

With this in mind,

$\Pr'(X \in I_p) = (1/n)\ \Sigma$ (over s=1 to m) of $\{N(I_s) \times [(f_Y(m(I_s)-m(I_p))\Pr(X \in I_p))]\ /\ [\Sigma$(over t=1 to m) of $(f_Y(m(I_s)-m(I_t))\Pr(X \in I_t))]$, where

I(x) is the interval in which "x" lies, $m(I_p)$ is the midpoint of the interval I_p, and $f(I_p)$ is the average value of the density function over the interval I_p, p=1,...m.

Using the preferred method of partitioning into intervals, the step at block 46 can be undertaken in $O(m^2)$ time. It is noted that a naive implementation of the last of the above equations will lead to a processing time of $O(m^3)$; however, because the denominator is independent of I_p, the results of that computation are reused to achieve $O(m^2)$ time. In the presently preferred embodiment, the number "m" of intervals is selected such that there are an average of 100 data points in each interval, with "m" being bound $10 \le m \le 100$.

It is next determined at decision diamond 48 whether the stopping criterion for the iterative process disclosed above has been met. In one preferred embodiment, the iteration is stopped when the reconstructed distribution is statistically the same as the original distribution as indicated by a X^2 goodness of fit test. However, since the true original distribution is not known, the observed randomized distribution (of the perturbed data) is compared with the [is compared with the] result of the current estimation for the reconstructed distribution, and when the two are statistically the same, the stopping criterion has been met, on the intuition that if these two are close, the current estimation for the reconstructed distribution is also close to the original distribution.

When the test at decision diamond 48 is negative, the integration cycle counter "j" is incremented at block 50, and the process loops back to block 46. Otherwise, the process ends at block 52 by returning the reconstructed distribution.

Now referring to Figure 5, the logic for constructing a decision tree classifier using the reconstructed distribution is seen. Commencing at block 54, for [reach] each attribute in the set "S" of data points, a DO loop is entered. Moving to block 56, split points for partitioning the data set "S" pursuant to growing the data tree are evaluated. Preferably, the split points tested are those between intervals, with each candidate split point being tested using the so-called "gini" index set forth in Classification and Regression Trees, Breiman et al., Wadsworth, Belmont, 1984. To summarize, for a data set S containing "n" classes (which can be predefined by the user, if desired) the "gini" index is given by $1-\Sigma p_j^2$, where p_j is the relative frequency of class "j" in the data set "S". For a split dividing "S" into subsets S1 and S2, the index of the split is given by:

index = $n_1/n(\text{gini}(S1)) + n_2/n(\text{gini}(S2))$, where n_1 = number of classes in S1 and n_2 = number of classes in S2.

The data points are associated with the intervals by sorting the values, and assigning the $N(I_1)$ lowest values to the first interval, the next highest values to the next interval, and so on.